

AMENDMENTS TO THE CLAIMS

This listing of claims will replace all prior versions, and listings, of claims in the application.

1-5. (Canceled)

6. (Currently Amended) A method for updating a running process, comprising:
allocating in executable program code text first memory space operable to
receive new program instructions comprising at least a first updated function:

allocating in executable program code text second memory space operable to
receive address information related to said new program instructions:

running said executable program code;

stopping execution of said executable program code;

injecting a jump instruction and an address of an update table at a location in a
memory containing a first instruction of a first replaced function, wherein said address of
said update table contains an address of a first instruction of said first updated function:

resuming execution of said executable program code, wherein said first updated
function is called in place of said first replaced function, and wherein said executable
code is updated in said memory;

determining a first distance between a position within said executable program code text
at which execution of said executable program code is stopped and an address of a first function,
wherein said first function is a function to be updated; and

in response to said first distance exceeding a predetermined amount, populating
an update table stored in memory with an address of ~~a~~the first updated function.

7. (Canceled)

8. (Previously Presented) The method of Claim 6, wherein said predetermined
amount is 8 bytes.

9-13. (Canceled)

14. (Currently Amended) A computer implemented method, the method comprising:
receiving information identifying:

a running executable program to be patched: and

a function to be replaced;
accessing a symbol table in a memory for said executable program to be patched;
obtaining from said symbol table an address of said function to be replaced;
stopping execution by a processor of said running executable program to be patched;
injecting in said running executable program to be patched at a location in said memory containing a first instruction of said function to be replaced a jump instruction and an address of a new function, wherein said new function is executed by said processor in place of said function to be replaced, and wherein a patched version of said executable program is created in said memory;
resuming execution of said patched version of said executable program by said processor;
~~wherein said patched version of said executable program is executed by said processor;~~ and
determining a number of bytes between a location within said executable program at which said executable program to be patched is stopped and an address of said function
to be replaced.

15. (Previously Presented) The method of Claim 14, further comprising:

in response to said determined number of bytes being at least as great as a first selected number, injecting in a stored copy of said running executable program to be patched said jump instruction and said address of said new function in place of said address of said function to be replaced, wherein a patched version of said executable program is created.

16-19. (Canceled)

20. (Original) A system for updating executing program code, comprising:

a create patch tool operable to receive information identifying an executable program to be updated and a function to be replaced;

a patch tool operable to query an operating system for a process identifier associated with said identified executable program;

a debugging utility operable to stop execution of said executable program to be updated and to determine a position of an instruction pointer associated with said executable program to be updated; and

a signal handler tool operable to replace in memory an address of said function to be replaced with an address of a replacement function in response to said position of said instruction pointer being at least a predetermined distance from said address of said replacement function, wherein said replacement function is executed instead of said function to be replaced upon resuming execution of said executable program, wherein said executable program is updated.

21. (Original) The system of Claim 20, wherein said signal handler is additionally operable to replace in a stored copy of said executable program an address of said function to be replaced with an address of a replacement function, wherein said stored copy of said executable program is updated.